Terms used **glyph** and **fragment** and **cached** and **bandwidth** Found **7,429** of **154,226**

Sort results by relevance

Display results expanded form

Save results to a Binder

Search Tips

Open results in a new window

Try an Advanced Search

Try this search in The ACM Guide

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next

Best 200 shown Relevance scale

1 Parallel texture caching

Homan Igehy, Matthew Eldridge, Pat Hanrahan

July 1999 **Proceedings of the ACM SIGGRAPH/EUROGRAPHICS workshop on Graphics hardware**

Full text available: pdf(1.80 MB) Additional Information: full citation, references, citings, index terms

2 Prefetching in a texture cache architecture

Homan Igehy, Matthew Eldridge, Kekoa Proudfoot

August 1998 **Proceedings of the ACM SIGGRAPH/EUROGRAPHICS workshop on Graphics hardware**

Full text available: pdf(1.45 MB) Additional Information: full citation, references, citings, index terms

3 The design and analysis of a cache architecture for texture mapping

Ziyad S. Hakura, Anoop Gupta

May 1997 **ACM SIGARCH Computer Architecture News , Proceedings of the 24th annual international symposium on Computer architecture**, Volume 25 Issue 2

Full text available: pdf(2.10 MB) Additional Information: full citation, abstract, references, citings, index terms

The effectiveness of texture mapping in enhancing the realism of computer generated imagery has made support for real-time texture mapping a critical part of graphics pipelines. Despite a recent surge in interest in three-dimensional graphics from computer architects, high-quality high-speed texture mapping has so far been confined to costly hardware systems that use brute-force techniques to achieve high performance. One obstacle faced by designers of texture mapping systems is the requirement ...

4 Semantic query caching in a mobile environment

Ken. C. K. Lee, H. V. Leong, Antonio Si

April 1999 **ACM SIGMOBILE Mobile Computing and Communications Review**, Volume 3 Issue 2

Full text available: pdf(1.41 MB) Additional Information: full citation, abstract, references, citings

Caching of remote data in a mobile client's local storage can improve data access performance and data availability. Traditional approaches are page-based, without taking advantage of the semantics of cached data. It is difficult for a client to determine if a query could be answered entirely based on locally cached data, forcing it to contact the database server for additional data. We propose a *semantic caching mechanism* which allows data to be cached as a collection of possibly related ...

5 The block-based trace cache
Bryan Black, Bohuslav Rychlik, John Paul Shen
May 1999 **ACM SIGARCH Computer Architecture News , Proceedings of the 26th annual international symposium on Computer architecture**, Volume 27 Issue 2
Full text available: pdf(181.08 KB) Additional Information: full citation, abstract, references, citings, index terms
Publisher Site

The trace cache is a recently proposed solution to achieving high instruction fetch bandwidth by buffering and reusing dynamic instruction traces. This work presents a new block-based trace cache implementation that can achieve higher IPC performance with more efficient storage of traces. Instead of explicitly storing instructions of a trace, pointers to blocks constituting a trace are stored in a much smaller trace table. The block-based trace cache renames fetch addresses at the basic block le ...

6 Proxy-based acceleration of dynamically generated content on the world wide web: An approach and implementation
Anindya Datta, Kaushik Dutta, Helen Thomas, Debra Vandermeer, Krithi Ramamritham
June 2004 **ACM Transactions on Database Systems (TODS)**, Volume 29 Issue 2
Full text available: pdf(927.23 KB) Additional Information: full citation, abstract, references, index terms

As Internet traffic continues to grow and websites become increasingly complex, performance and scalability are major issues for websites. Websites are increasingly relying on dynamic content generation applications to provide website visitors with dynamic, interactive, and personalized experiences. However, dynamic content generation comes at a cost---each request requires computation as well as communication across multiple components.To address these issues, various dynamic content caching ap ...

Keywords: Edge caching, caching dynamically generated content, fragment caching, implementation, proxy caching, world wide web

7 Research sessions: distributed systems: Proxy-based acceleration of dynamically generated content on the world wide web: an approach and implementation
Anindya Datta, Kaushik Dutta, Helen Thomas, Debra VanderMeer, Suresha, Krithi Ramamritham
June 2002 **Proceedings of the 2002 ACM SIGMOD international conference on Management of data**
Full text available: pdf(1.37 MB) Additional Information: full citation, abstract, references, citings, index terms

As Internet traffic continues to grow and web sites become increasingly complex, performance and scalability are major issues for web sites. Web sites are increasingly relying on dynamic content generation applications to provide web site visitors with dynamic, interactive, and personalized experiences. However, dynamic content generation comes at a cost --- each request requires computation as well as communication across multiple components.To address these issues, various dynamic content cach ...

Keywords: dynamic content, edge caching, proxy-based caching

8 Neon: a single-chip 3D workstation graphics accelerator
Joel McCormack, Robert McNamara, Christopher Gianos, Larry Seiler, Norman P. Jouppi, Ken Correll
August 1998 **Proceedings of the ACM SIGGRAPH/EUROGRAPHICS workshop on Graphics hardware**
Full text available: pdf(1.58 MB) Additional Information: full citation, references, citings, index terms

Keywords: chunk rendering, direct rendering, graphics pipeline, level of detail,

rasterization, texture cache, tile rendering

9 Efficient use of memory bandwidth to improve network processor throughput

Jahangir Hasan, Satish Chandra, T. N. Vijaykumar

May 2003 **ACM SIGARCH Computer Architecture News , Proceedings of the 30th annual international symposium on Computer architecture**, Volume 31 Issue 2

Full text available: pdf(184.83 KB) Additional Information: full citation, abstract, references

We consider the efficiency of packet buffers used in packet switches built using network processors (NPs). Packet buffers are typically implemented using DRAM, which provides plentiful buffering at a reasonable cost. The problem we address is that a typical NP workload may be unable to utilize the peak DRAM bandwidth. Since the bandwidth of the packet buffer is often the bottleneck in the performance of a shared-memory packet switch, inefficient use of available DRAM bandwidth further reduces th ...

10 Improving instruction cache behavior by reducing cache pollution

Rajiv Gupta, Chi-Hung Chi

November 1990 **Proceedings of the 1990 ACM/IEEE conference on Supercomputing**

Full text available: pdf(1.01 MB) Additional Information: full citation, abstract, references

In this paper we describe compiler techniques for improving instruction cache performance. Through repositioning of the code in main memory, leaving memory locations unused, code duplication, and code propagation, the effectiveness of the cache can be improved due to reduced cache pollution and fewer cache misses. Results of experiments indicate that significant reduction in bus traffic results from the use of these techniques. Since memory bandwidth is a critical resource in shared memory multi ...

Keywords: cache misses, cache pollution, control dependence graph, control flow graph, instruction prefetching, program dependence graph

11 Enhancing Multimedia Caching Algorithm Performance Through New Interval Definition Strategies

Javier Fernández, Jesus Carretero, Felix Garcia, Jose M. Perez, A. Calderon

March 2003 **Proceedings of the 36th annual symposium on Simulation**

Full text available: pdf(392.54 KB) Additional Information: full citation, abstract, index terms

Nowadays, multimedia systems are evolving towards integratedstorage platforms that meet the requirements ofdeterministic applications, multimedia systems, and traditionalbest-effort applications altogether. These systemsmust incorporate a disk scheduling mechanism and a cachearchitecture that can handle the requirements of each kindof request while showing a good overall performance. Inthis paper a new interval caching strategy is proposed thatincludes several optimizations to the state of the a ...

12 Ray tracing on programmable graphics hardware

Timothy J. Purcell, Ian Buck, William R. Mark, Pat Hanrahan

July 2002 **ACM Transactions on Graphics (TOG) , Proceedings of the 29th annual conference on Computer graphics and interactive techniques**, Volume 21 Issue 3

Full text available: pdf(454.93 KB) Additional Information: full citation, abstract, references, citings, index terms

Recently a breakthrough has occurred in graphics hardware: fixed function pipelines have been replaced with programmable vertex and fragment processors. In the near future, the graphics pipeline is likely to evolve into a general programmable stream processor capable of more than simply feed-forward triangle rendering.In this paper, we evaluate these trends in programmability of the graphics pipeline and explain how ray tracing can be mapped to graphics hardware. Using our simulator, we analyze ...

Keywords: programmable graphics hardware, ray tracing

13 The Zebra striped network file system
John H. Hartman, John K. Ousterhout
August 1995 **ACM Transactions on Computer Systems (TOCS)**, Volume 13 Issue 3

Full text available: pdf(2.76 MB) Additional Information: full citation, abstract, references, citings, index terms, review

Zebra is a network file system that increases throughput by striping the file data across multiple servers. Rather than striping each file separately, Zebra forms all the new data from each client into a single stream, which it then stripes using an approach similar to a log-structured file system. This provides high performance for writes of small files as well as for reads and writes of large files. Zebra also writes parity information in each stripe in the style of RAID disk arrays; this ...

Keywords: RAID, log-based striping, log-structured file system, parity computation

14 Fbufs: a high-bandwidth cross-domain transfer facility
Peter Druschel, Larry L. Peterson
December 1993 **ACM SIGOPS Operating Systems Review , Proceedings of the fourteenth ACM symposium on Operating systems principles**, Volume 27 Issue 5

Full text available: pdf(1.35 MB) Additional Information: full citation, abstract, references, citings, index terms

We have designed and implemented a new operating system facility for I/O buffer management and data transferacross protection domain boundaries on shared memory machines. This facility, called *fast buffers* (fbufs), combines virtual page remapping with shared virtual memory, and exploits locality in I/O traffic to achieve high throughput without compromising protection, security, or modularity. goal is to help deliver the high bandwidth afforded by emerging high-speed networks to user-leve ...

15 The Zebra striped network file system
John H. Hartman, John K. Ousterhout
December 1993 **ACM SIGOPS Operating Systems Review , Proceedings of the fourteenth ACM symposium on Operating systems principles**, Volume 27 Issue 5

Full text available: pdf(1.93 MB) Additional Information: full citation, abstract, references, citings, index terms

Zebra is a network file system that increases throughput by striping file data across multiple servers. Rather than striping each file separately, Zebra forms all the new data from each client into a single stream, which it then stripes using an approach similar to a log-structured file system. This provides high performance for writes of small files as well as for reads and writes of large files. Zebra also writes parity information in each stripe in the style of RAID disk arrays; this increase ...

16 Delay streams for graphics hardware
Timo Aila, Ville Miettinen, Petri Nordlund
July 2003 **ACM Transactions on Graphics (TOG)**, Volume 22 Issue 3

Full text available: pdf(1.67 MB) Additional Information: full citation, abstract, references, index terms

In causal processes decisions do not depend on future data. Many well-known problems, such as occlusion culling, order-independent transparency and edge antialiasing cannot be properly solved using the traditional causal rendering architectures, because future data may change the interpretation of current events.We propose adding a *delay stream* between the vertex and pixel processing units. While a triangle resides in the delay stream, subsequent triangles generate occlusion information. ...

Keywords: 3D graphics hardware, antialiasing, occlusion culling, order-independent

transparency, stream processing

17 Improving trace cache effectiveness with branch promotion and trace packing
Sanjay Jeram Patel, Marius Evers, Yale N. Patt
April 1998 **ACM SIGARCH Computer Architecture News , Proceedings of the 25th annual international symposium on Computer architecture**, Volume 26 Issue 3
Full text available: pdf(1.11 MB) Publisher Site Additional Information: full citation, abstract, references, citings, index terms

The increasing widths of superscalar processors are placing greater demands upon the fetch mechanism. The trace cache meets these demands by placing logically contiguous instructions in physically contiguous storage. As a result, the trace cache delivers instructions at a high rate by supplying multiple fetch blocks each cycle.In this paper, we examine two techniques to improve the number of instructions delivered each cycle by the trace cache. The first technique, branch promotion, dynamically ...

18 Versioning and fragmentation: Automatic detection of fragments in dynamically generated web pages
Lakshmish Ramaswamy, Arun Iyengar, Ling Liu, Fred Douglis
May 2004 **Proceedings of the 13th international conference on World Wide Web**
Full text available: pdf(268.12 KB) Additional Information: full citation, abstract, references, index terms

Dividing web pages into fragments has been shown to provide significant benefits for both content generation and caching. In order for a web site to use fragment-based content generation, however, good methods are needed for dividing web pages into fragments. Manual fragmentation of web pages is expensive, error prone, and unscalable. This paper proposes a novel scheme to automatically detect and flag fragments that are cost-effective cache units in web sites serving dynamic content. We consider ...

Keywords: L-P fragments, dynamic content caching, fragment detection, fragment-based caching, shared fragments

19 A general framework for prefetch scheduling in linked data structures and its application to multi-chain prefetching
Seungryul Choi, Nicholas Kohout, Sumit Pamnani, Dongkeun Kim, Donald Yeung
May 2004 **ACM Transactions on Computer Systems (TOCS)**, Volume 22 Issue 2
Full text available: pdf(2.45 MB) Additional Information: full citation, abstract, references, index terms

Pointer-chasing applications tend to traverse composite data structures consisting of multiple independent pointer chains. While the traversal of any single pointer chain leads to the serialization of memory operations, the traversal of independent pointer chains provides a source of memory parallelism. This article investigates exploiting such *interchain memory parallelism* for the purpose of memory latency tolerance, using a technique called *multi--chain prefetching*. Previous work ...

Keywords: Data prefetching, memory parallelism, pointer-chasing code

20 Consistency and replication: Evaluation of edge caching/offloading for dynamic content delivery
Chun Yuan, Yu Chen, Zheng Zhang
May 2003 **Proceedings of the twelfth international conference on World Wide Web**
Full text available: pdf(161.49 KB) Additional Information: full citation, abstract, references, citings, index terms

As dynamic content becomes increasingly dominant, it becomes an important research topic as how the edge resources such as client-side proxies, which are otherwise underutilized for

such content, can be put into use. However, it is unclear what will be the best strategy and the design/deployment tradeoffs lie therein. In this paper, using one representative e-commerce benchmark, we report our experience of an extensive investigation of different offloading and caching options. Our results point ...

Keywords: dynamic content, edge caching, offloading

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next